

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method for maintaining a system for database management, the method comprising:
 1. during splitting of a leaf block of a database index recording an address of a newly created leaf block; [[and]]
 1. maintaining the [[new]] address of the newly created leaf block in a list as part of metadata of a primary B+tree [[.]] ; and
 2. maintaining a measure of invalid guess-database block addresses by calculating a ratio of a count of database block addresses in the list of new addresses to a total number of leaf blocks of the primary B+tree.

A3
2. (cancelled)

2
3. (currently amended) The method according to claim [[2]] 1, wherein the measure of invalid guess-database block addresses applies to mapping tables and secondary indexes on the primary B+tree.

3
4. (currently amended) The method according to claim [[2]] 1, wherein the list of database block addresses and the ratio are maintained only when the ratio is less than a threshold value.

4
5. (original) The method according to claim 4, wherein the threshold value for the ratio is about 10%.

5
6. (original) The method according to claim 5, further comprising:
adjusting a guess-DBA quality of at least one of the mapping table and the secondary index utilizing the ratio.

6
7. (original) The method according to claim 4, wherein if the ratio is below the threshold value the method further comprises:
selectively correcting entries in the mapping table and/or secondary index.

7
8. (original) The method according to claim 7, wherein correcting entries in the mapping table comprises for all rows in a list of blocks in the primary B+tree:
obtaining corresponding mapping table row identifiers and database block addresses of a current block in the list;
sorting the corresponding mapping table row identifiers;

obtaining mapping table rows corresponding to the mapping table row
identifiers; and

updating a guess-DBA component if it has changed.

8

1

9. (original) The method according to claim 8, wherein the correcting is
carried out on-line in a piece-wise manner.

9

3

10. (original) The method according to claim 4, wherein correcting entries
in the secondary index comprises for all rows in a list of blocks in the primary
B+tree:

obtaining a secondary index key, a primary key and a database block
address of a current block in the list of blocks;

11. (original) The method according to claim 4, wherein
sorting the secondary index keys, primary keys and database addresses in
order of (secondary index key, primary key) pairs;

obtaining an index row corresponding to the (secondary index key, primary
key) pair; and

12. (original) The method according to claim 4, wherein
updating a guess-DBA component of the index row if the guess-DBA has
changed.

10

9

13. (currently amended) The method according to claim [[11]] 10, wherein
the correcting is carried out on-line in a piece-wise manner.

11 3
12. (original) The method according to claim 4, wherein if the ratio is
above the threshold value the method further comprises:

correcting guess-database addresses on a per object basis.

12 11
13. (original) The method according to claim 12, wherein correcting guess-
database block addresses on the mapping table comprises:

performing a full scan of the mapping table;

determining for each row of the mapping table a correct guess-database
block address by traversing the primary B+tree up to a penultimate level;

updating each row of the mapping table with the correct guess-database
block address; and

committing the correct guess-database address to the mapping table in
batches.

13 11
14. (original) The method according to claim 12, wherein correcting guess-
database block addresses on a per object basis comprises for each secondary index
object:

performing a full scan of the secondary index object;
determining for each row of the secondary index a correct guess-database
block address by traversing the primary B+tree up to a penultimate level;

updating each row of the secondary index with the correct guess-database block address; and

committing the correct guess-database block address to the secondary index in batches.

4

~~15.~~ (original) The method according to claim 1, further comprising:
maintaining a list of database block addresses in the list.

5

~~16.~~ (currently amended) A system ~~for organizing a database index, the~~ comprising:
system

a list of addresses of blocks newly created during splitting of a primary
B+tree [.] ;

a count of database block addresses in the list; and

a ratio of count of database block addresses to total number of leaf blocks

as a measure of invalid guess-database block addresses.

17. (cancelled)

18. (cancelled)

16
19. (original) The system according to claim 16, wherein the database index
is a primary B+tree structure, wherein the system further comprises:

a mapping table used to support bitmap indexes.

11

16
20. (original) The system according to claim 19, further comprising:

a bitmap index supported by the mapping table.

16

15
21. (original) The system according to claim 16, wherein the database index
is a primary B+tree structure, wherein the system further comprises:

a secondary index structure comprising hybrid row identifiers.

19

22. (currently amended) A computer program product for performing a
process for maintaining a database management system, comprising:

a computer readable medium; and

computer program instructions, recorded on the computer readable
medium, executable by a processor, for performing the steps of:

during splitting of a leaf block of a primary B+tree recording an address of
a newly created leaf block; [[and]]

maintaining the [[new]] address of the newly created leaf block in a list as
part of primary B+tree metadata [[.]] ; and

maintaining a measure of invalid guess-database block addresses by
calculating a ratio of a count of database block addresses in the list of new
addresses to a total number of leaf blocks of the primary B+tree.

20

23. (currently amended) A system for performing a process for maintaining a database management system, comprising:
a processor operable to execute computer program instructions; and
a memory operable to store computer program instructions executable by the processor, for performing the steps of:

during splitting of a leaf block of a primary B+tree recording an address of a newly created leaf block; [[and]]

maintaining the [[new]] address of the newly created leaf block in a list as part of primary B+tree metadata [[.]] ; and

maintaining a measure of invalid guess-database block addresses by
calculating a ratio of a count of database block addresses in the list of new
addresses to a total number of leaf blocks of the primary B+tree.